Machine Learning Based Feature Reduction for Network Intrusion Detection

Abdelshakour Abuzneid, Miad Faezipour, Razan Abdulhammed, Arafat AbuMallouh and Hassan Musafer

Department of Computer Science and Engineering

University of Bridgeport, Bridgeport, CT 06604, USA

Abstract

The security of networked systems has become a critical universal issue. The rate of attacks against networked systems has increased dramatically, and the tactics used by the attackers are continuing to evolve. Intrusion detection is one of the solutions against these attacks. A common and effective approach for designing Intrusion Detection Systems (IDS) is Machine Learning. The performance of an IDS is significantly improved when the features are more discriminative and representative. This study uses two feature dimensionality reduction approaches: i) Auto-Encoder (AE): an instance of deep learning, for dimensionality reduction, and ii) Principle Component Analysis (PCA). The resulting low-dimensional features from both techniques are then used to build various classifiers such as Random Forest (RF), Bayesian Network, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) for designing an IDS. The experimental findings with low-dimensional features in binary and multi-class classification show better performance in terms of Detection Rate (DR), F-Measure, False Alarm Rate (FAR), and Accuracy. This research effort is able to reduce the CICIDS2017 dataset's feature dimensions from 81 to 10, while maintaining a high accuracy of 99.6%. Furthermore, we propose a Multi-Class Combined performance metric $Combined_{Mc}$ with respect to class distribution to compare various multi-class and binary classification systems through incorporating FAR, DR, Accuracy, and class distribution parameters. In addition, we developed a uniform distribution based balancing approach to handle the imbalanced distribution of the minority class instances in the CICIDS2017 network intrusion dataset.

Network Intrusion Detection

- Network Intrusion Detection System (IDS) is a software-based application or a hardware device that is used to identify malicious behavior in the network.
  - Anomaly-based
    - **Machine learning (ML) techniques can predict and detect threats before they result in major network security incidents**.
      - **Feature Representation and Extraction**
      - **Classification**
  - Signature-based
- The performance of an IDS is significantly improved when the features are more discriminative and representative.
  - **Feature Reduction**
- Network traffic is heavily nonlinear and imbalanced

Proposed Framework

The procedure of our proposed framework, as presented in Figure 1, mainly includes Preprocessing, Unity-Based Normalization, Dimensionality reduction, Classification and Evaluation, and finally, combating Imbalanced class distributions using the uniform distribution based balance approach.

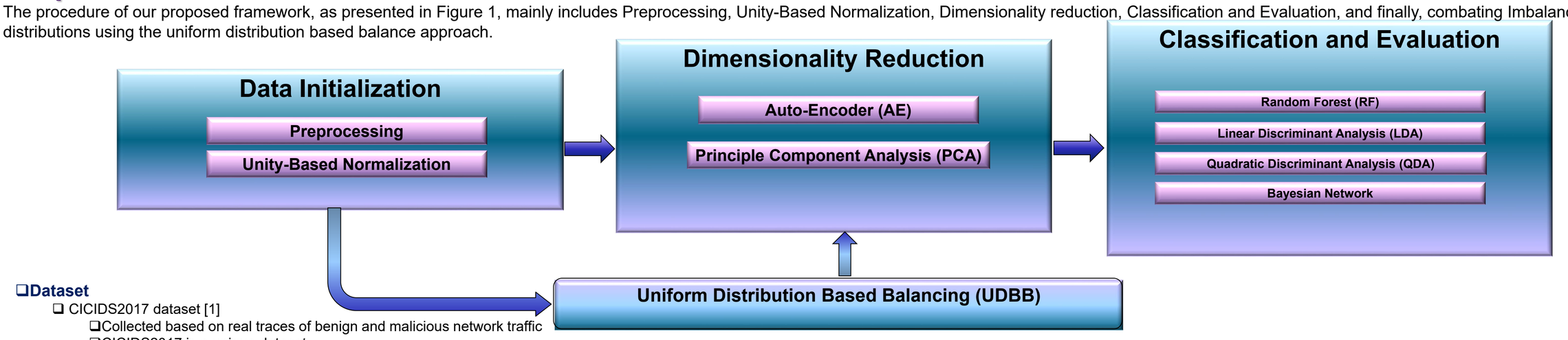

Figure 1. Proposed IDS Framework

Dataset

- CICIDS2017 dataset [1]
  - Collected based on real traces of benign and malicious network traffic
  - CICIDS2017 is a unique dataset
  - Includes up-to-date attacks
  - Features are exclusive and matchless in comparison with other datasets such as AWID [2,3,4], and CIDD-001 [5]
  - Selected as most comprehensive IDS benchmark

Data Initialization

- Preprocessing: Mapping IP address to an integer representation
- Unity Based Normalization: Rescaling

Auto-Encoder (AE)

- Input vector $x = (x_1, x_2, \dots, x_n)$ is first compressed to a lower dimensional hidden representation that consists of one or more hidden layers $a = (a_1, a_2, \dots, a_n)$.
- The hidden representation a is then mapped to reproduce the output $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$.

Dimensionality Reduction using Auto-Encoder (AE)

- Perform feed-forward pass on all training instances
- Compute the output, sparsity mean and error of the cost function
- Compute the cost function of the sparse auto-encoder
- Back-propagate the error to update the weights and the bias for all the layers
- Compute the reduced features from the hidden layer
- A two hidden layer sparse auto-encoder is used with sigmoid activation functions and tied weights.

Principle Component Analysis (PCA)

- A projection-based mechanism
- The original dataset with n columns (features) is projected into a subspace with k or lower dimensions representation, whilst retaining the essence of the original data.
- The data is first preprocessed to normalize its mean and variance.
- Then, the covariance matrix, Eigen-vectors and Eigen-values are calculated.

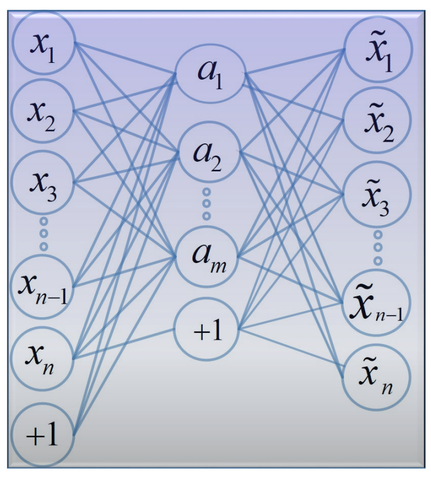

Figure 2. Structure of an Auto-Encoder

Proposed $Combined_{Mc}$ Calculation Pseudo-Code

Feed Confusion Matrix (CM)
For i =1 to C
Calculate the total number of False Positive (FP) for C_i as the sum of values in the i^{th} column excluding TP
Calculate the total number of False Negatives (FN) for C_i as the sum of values in the i^{th} row excluding TP
Calculate the total number of True Negatives (TN) for C_i as the sum of all columns and rows excluding the i^{th} row and column
Calculate the total number of True Positives (TP) for C_i as the diagonal of the i^{th} cell of CM
Calculate the total number of instances for C_i as the sum of the i^{th} row
Calculate the total number of instances in the dataset as the sum of all rows
Calculate Accuracy (Acc), Detection Rate (DR), and False Alarm Rate (FAR) for each class C_i
Calculate the distribution of each C_i : $\lambda_i = \frac{Number\ of\ Instances\ in\ Class\ i}{Number\ of\ Instances\ in\ the\ dataset}$
i ++
Calculate $Combined_{MC} = \sum_{i=1}^{C} \lambda_i \left(\frac{Acc_i + DRi}{2} - FAR_i \right)$

Proposed UDBB Pseudo-Code

Input Training Set: D_{Train}
Set Distribution to Uniform
C : Number of Classes
F_T : Total number of features in D_{Train} Training Set
I_{old} : Total number of Instances in D_{Train}
Calculate the required number of Instances in each class: $I_{resample}$, where $I_{resample} = \frac{Number\ of\ Instances\ in\ the\ dataset}{Number\ of\ Classes\ in\ the\ dataset}$
Training Set $DTrain_{new} = \emptyset$
For each class C_i, Do
While $i \neq I_{Resample}$
For each feature $F_1, \dots, F_T$
Generate new sample using uniform distribution
Assign Class label
Return $Dtrain_{new}$

Results

- Proposed ideas tested on the CICIDS2017 dataset
- Random Forest classifier with reduced features using AE and PCA show significantly high performances.
  - Binary and Multi-class classification
  - UDBB applied

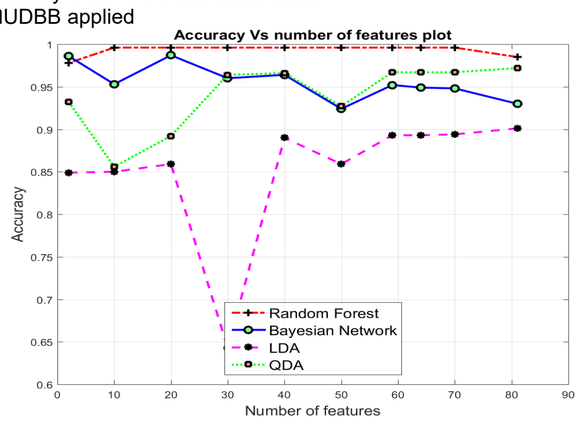

Figure 3. Accuracy in terms of number of components using PCA

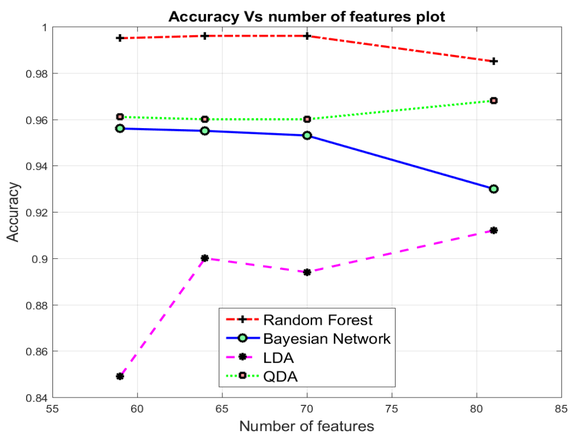

Figure 4. Accuracy in terms of number of features using Auto-Encoder

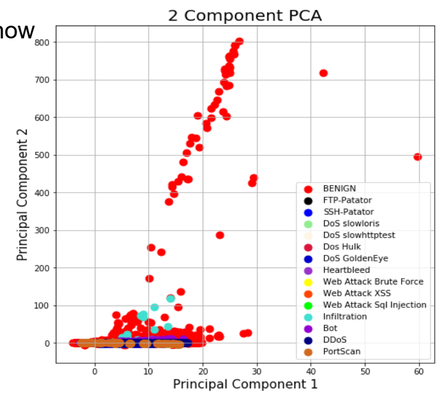

Figure 5. PCA visualization before applying UDBB

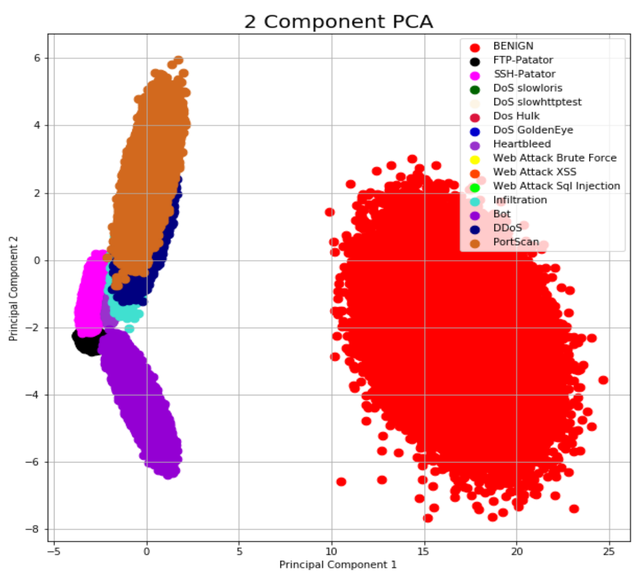

Figure 6. PCA visualization after applying UDBB

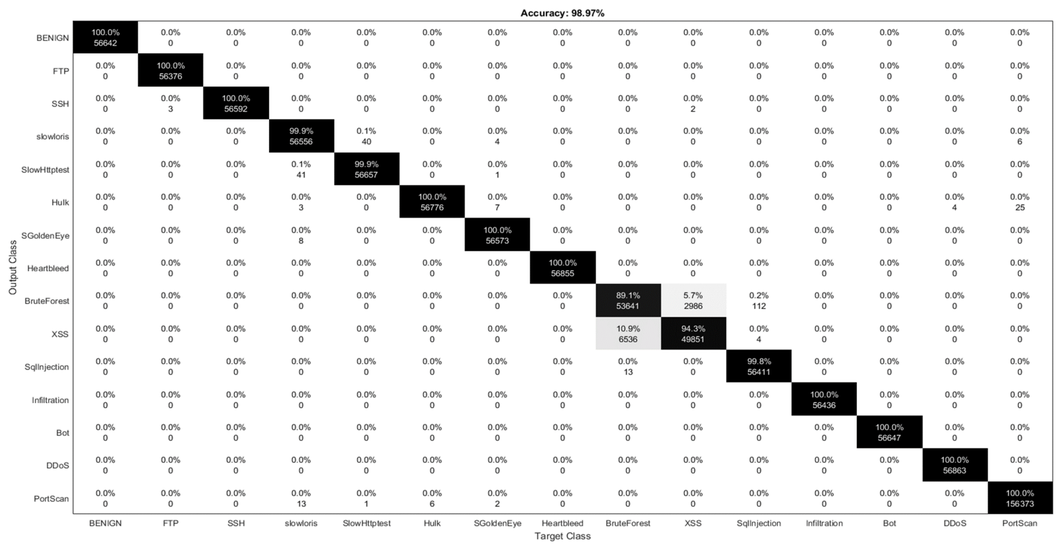

Figure 7. Confusion Matrix for (PC A− RF) Mc−10 After Applying UDBB Approach

Table 1. Comparison with previous related work

Reference	Classifier name	F-measure	Feature selection/extraction (Features Count)
[6]	MLP	0.948	Payload related features
[7]	SVM	0.921	DBN
[8]	KNN	0.997	Fisher Scoring (30)
[9]	XGBoost for DoS Attacks	0.995	(80)
[10]	Deep Learning for Port Scan Attacks	Accuracy 97.80	(80)
	SVM for Port Scan Attacks	Accuracy 69.79	(80)
[11]	XGBoost	Accuracy 98.93	DDR Features Selections (36)
[12]	Deep Multi Layer Perceptron (DMLP) for DDoS Attacks	Accuracy 91.00	Recursive feature elimination with Random Forest
Proposed Framework	**Random Forest**	**0.995**	**Auto-encoder (59)**
Proposed Framework	**Random Forest**	**0.996**	**PCA with Original Distribution (10)**
Proposed Framework	**Random Forest**	**0.988**	**PCA With UDBB (10)**

References and Published Work

[1] Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization,"in Proceedings of fourth international conference on information systems security and privacy, ICISSP , 2018.

[2] R. Abdulhammed, M. Faezipour, A. Abuzneid and A. AbuMallouh, "Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic", ***IEEE Sensors Letters, Special Issue on Multimodal Data Fusion, Vol. 3, No. 1, pp. 1-4,*** **Jan. 2019.**

[3] R. Abdulhammed, M. Faezipour, A. Abuzneid and A. Alessa, "Enhancing Wireless Intrusion Detection Using Machine Learning Classification with Reduced Attribute Sets", ***in Proceedings of the IEEE International Wireless Communications and Mobile Computing Conference (IEEE IWCMC 2018), pp. 524-529,*** **Limassol, Cyprus, June 2018.**

[4] R. Abdulhammed, M. Faezipour, A. Abuzneid and A. Alessa, "Effective Features Selection and Machine Learning Classifiers for Improved Wireless Intrusion Detection", ***in Proceedings of the IEEE International Symposium on Networks, Computers and Communications (IEEE ISNCC 2018), pp. 1-6,*** **Rome, Italy, June 2018.**

[5] R. Abdulhammed, M. Faezipour, and K. Elleithy, Intrusion Detection in Self organizing Network:A Survey NewYork: CRC Press Taylor Francis Group, 2017, ch. 13, pp. 393–449.

[6] Watson, G. A Comparison of Header and Deep Packet Features when Detecting Network Intrusions. Technical report, 2018.

[7] Marir, N.; Wang, H.; Feng, G.; Li, B.; Jia, M. Distributed Abnormal Behavior Detection Approach based on Deep Belief Network and Ensemble SVM using Spark. IEEE Access 2018

[8] Aksu, D.; Üstebay, S.; Aydin, M.A.; Atmaca, T. Intrusion Detection with Comparative Analysis of Supervised Learning Techniques and Fisher Score Feature Selection Algorithm. International Symposium on Computer and Information Sciences. Springer, 2018, pp. 141–149.

[9] Bansal, A.; Kaur, S. Extreme Gradient Boosting Based Tuning for Classification in Intrusion Detection Systems. International Conference on Advances in Computing and Data Sciences. Springer, 2018, pp. 372-380.

[10] Aksu, D.; Aydin, M.A. Detecting Port Scan Attempts with Comparative Analysis of Deep Learning and Support Vector Machine Algorithms, IEEE IBIGDELFT, 2018, pp. 77–80.

[11] Bansal, A. DDR Scheme and LSTM RNN Algorithm for Building an Efficient IDS. Master's thesis, 2018.

[12] Ustebay, S.; Turgut, Z.; Aydin, M.A. Intrusion Detection System with Recursive Feature Elimination by Using Random Forest and Deep Learning Classifier. IEEE IBIGDELFT, 2018, pp. 71–76.

Emerging Communications Technologies Research Center

D-BEST Lab — Digital / Biomedical Embedded Systems & Technology Lab